

COURSE PROJECT

VLSI

Yash Nitin Dusane
2022102078

QUESTION:

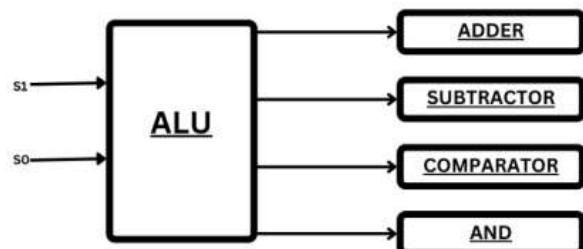
VLSI Project Monsoon 2023

Design an ALU that can perform a 4-Bit addition, subtraction, comparison, ANDing. Estimate the critical path, maximum delay possible in the circuit. Design the layout of your ALU, clearly indicate the location of each standard cell in the design. Compare your pre- and post-layout results. Also, Verify the functionality of your ALU using Verilog.

Tools that can be used:

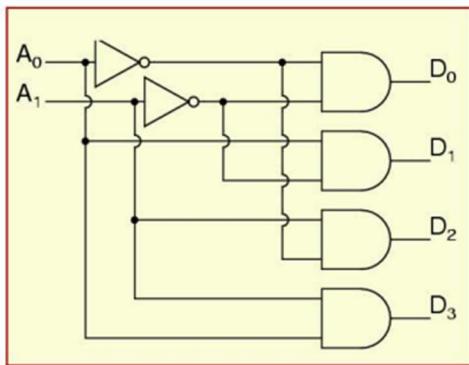
- NG-SPICE for circuit design.
- Magic for Layout.
- Verilog.

Block Diagram:



Thus, the question demands us to make an ALU which performs these 4 operations, so we are required to make 2 to 4 Decoder along with the whole circuit so that the 2 select inputs operate to give us the desired output.

2 TO 4 DECODER :

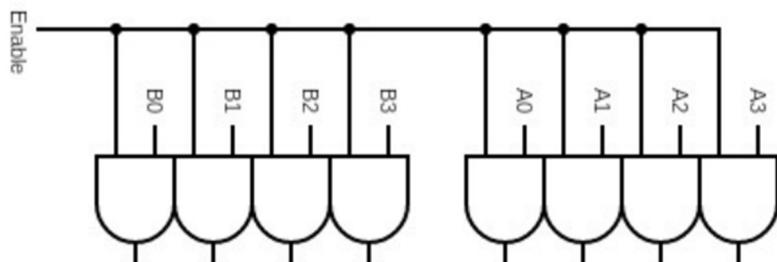


S1 S0 operation

0 0	Add
0 1	Subtract
1 0	Compare
1 1	And

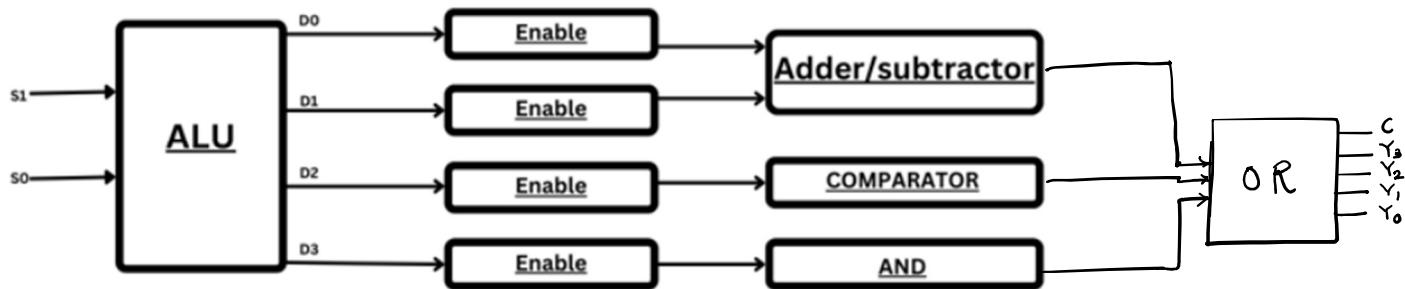
Hence, output will accordingly display according to the select lines when the decoder output acts as an enablers for the 4 operations, when one of the outputs of decoder is 1, others are 0, so the inputs connecting to all other inputs should be 0.

ENABLER BLOCK :



So, we connect this Enabler block before passing each input to the 4 operation subcircuits.

FINAL CIRCUIT IMPLEMENTATION :



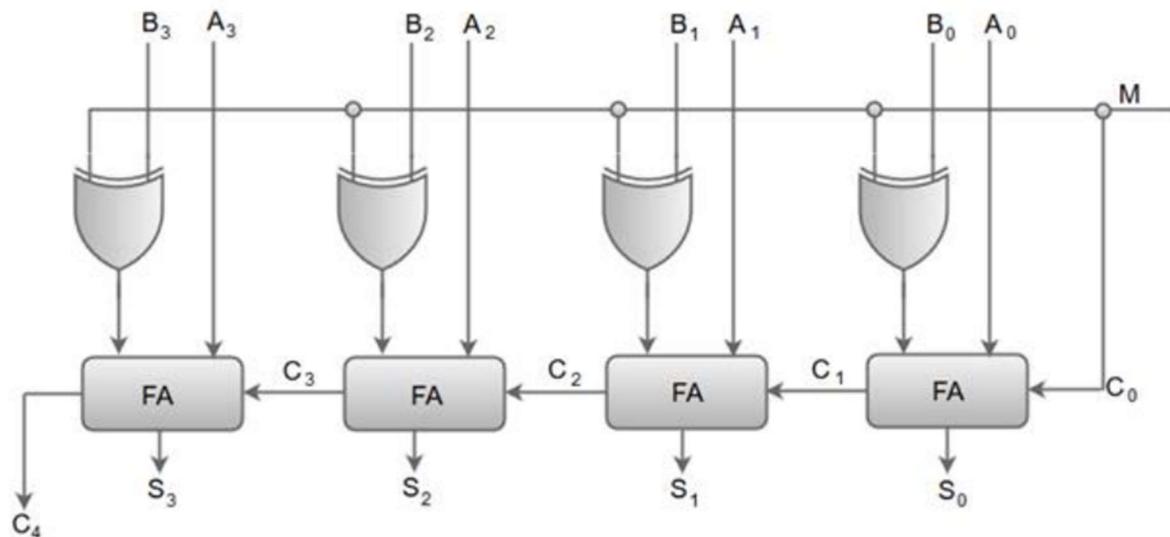
Also instead of making Adder, subtractor separated, I can implement the combined circuit of Adder-Subtractor. At last we get 5(Adder-Subtractor) + 3(Comparator) + 4(AND) outputs. Any of the one set would only show up result as others would be enabled 0. Thus, we require at maximum a 5-bit output. So we can just pass all the output bits through OR gate and get the output.

ASSUMPTION:

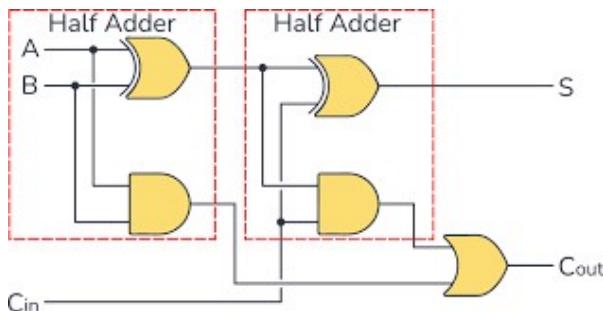
1. When the output requires less than 5 bits, the remaining bits would be 0 by default.
2. The output format would be MSB to LSB as left to right. So the 4 bit adder output would have the 1st left MSB bit by default as 0. Similarly for COMPARATOR, the 2 MSB bits would be by default 0.

ADDER SUBTRACTOR BLOCK :

4 bit adder-subtractor:



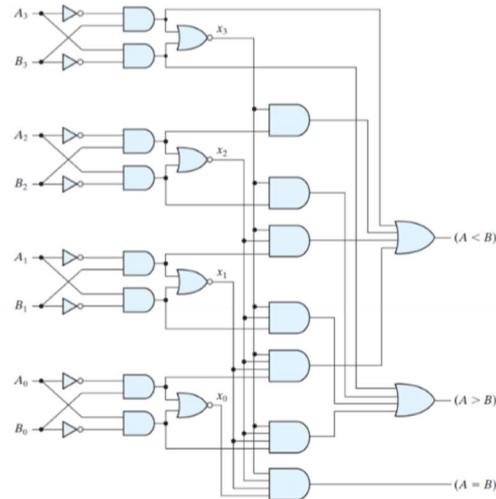
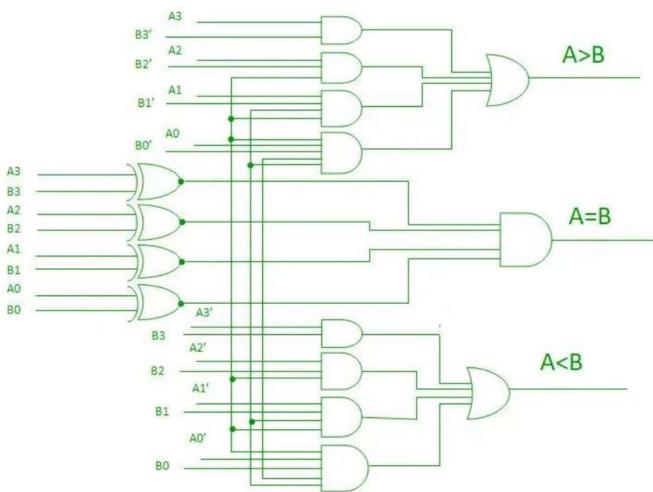
FULL ADDER CIRCUIT :



Thus, the XOR gate implementation gives us the combined circuit where M acts as carry bit as well as the bit being XOR with input bits. So for M=0, bits don't change and no carry bit Co, while for M=1, the inputs are inverted and 1 is added as carry bit so we get 2's complement of B added to other input A which becomes A-B.

- Here, select input S0 is used as M.
- In case of subtractor, the MSB bit C4 acts like a sign bit for subtraction (0 for negative, 1 for positive).
- As Adder and Subtractor needs both D0 and D1 as enables, so D0 and D1 are needed to be OR before being used to enable the Adder-Subtractor block.

COMPARATOR BLOCK



The comparator bit uses XNOR gates for equality of bits and AND gates to cover the rest of the cases of lesser or greater by AND of one complement with other and equality of bits preceding it.

$$x_i = A_i B_i + A'_i B'_i \text{ for } i = 0, 1, 2, 3$$

$$(A = B) = x_3 x_2 x_1 x_0$$

$$(A > B) = A_3 B'_3 + x_3 A_2 B'_2 + x_3 x_2 A_1 B'_1 + x_3 x_2 x_1 A_0 B'_0$$

$$(A < B) = A'_3 B_3 + x_3 A'_2 B_2 + x_3 x_2 A'_1 B_1 + x_3 x_2 x_1 A'_0 B_0$$

- Here the output for the comparator would be the 3 bits from LSB in 5 bit output in form A>B, A=B, A<B from right to left. Also as mentioned the 2 MSB bits would be always 0.
- When the enable is 0, we could see A=B resulting as 1, so to avoid it the, output A=B was needed to be AND with its enable from 2 to 4 Decoder, so that it doesn't give output when enable is 0.

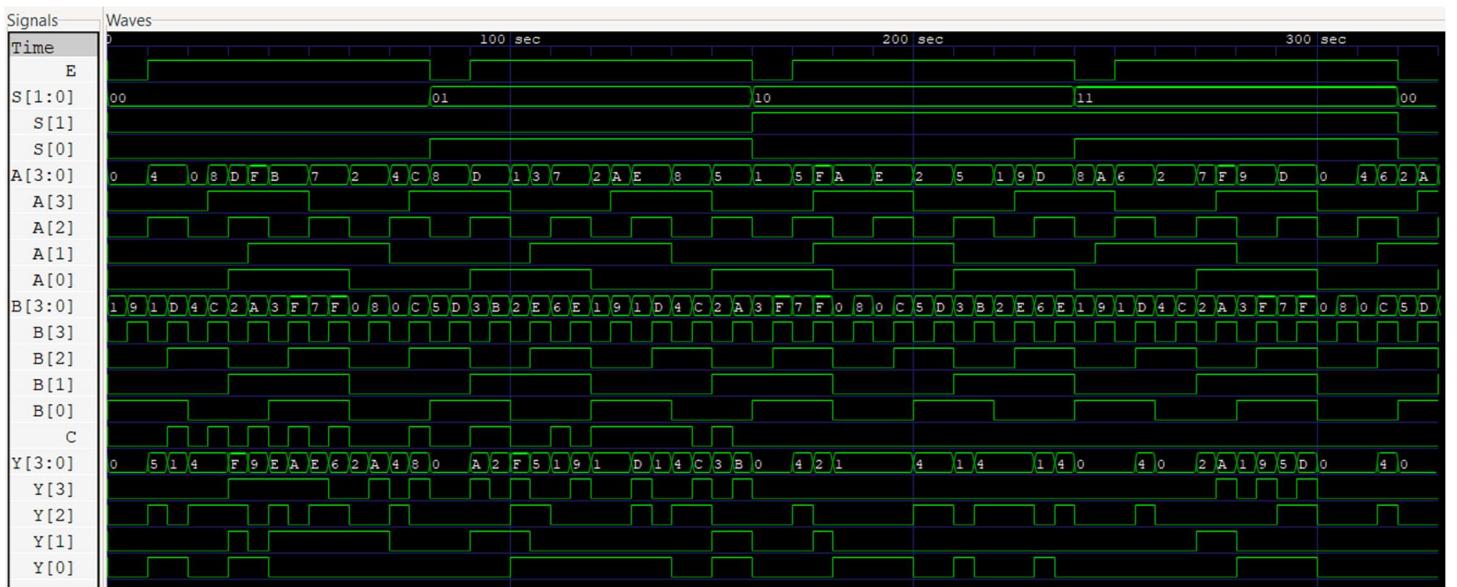
VERILOG IMPLEMENTATION

To make a Structural Code of the circuit, just we need to make the circuit modules of ADDERSUBTRACTOR, COMPARATOR, AND 4BITAND, using the inbuilt gate modules which work in format : (example of AND gate implementation)
AND A1(Output, Input1, Input2, ...).
(Also using array as In[3:0] to make a 4 bit input named In.)

After making all modules, combining the whole circuit into one ALU (named 2TO4DECODER) circuit. Initially code for 2 to 4 Decoder circuit and then the output of the Decoder to be used as enable while calling the next modules). So at last after getting all the outputs, combining them into 5 bit output using OR gate module.

Hence, after the ALU circuit is implemented in the Verilog, it just needs to be tested by calling the module in testbench and testing it for various outputs.

GTKWave Output :



Output when a single input is given :

↙ TERMINAL

Truth Table based on Structural Code of ALU Circuit :

E = 0, S = 00, A = 1111, B = 1111,
4bit Output = 0000, Carry Output = 0

E = 1, S = 00, A = 1111, B = 1111,
4bit Output = 1110, Carry Output = 1
Thus, 4 bit Adder Output = 1110

E = 1, S = 01, A = 1111, B = 1111,
4bit Output = 0000, Carry Output = 1
Thus, 4bit Subtractor Output = 10000

E = 1, S = 10, A = 1111, B = 1111,
4bit Output = 0010, Carry Output = 0
Thus, 4 bit Comparator Output : A>B = 0, A=B = 1, A<B = 0

E = 1, S = 11, A = 1111, B = 1111,
4bit Output = 1111, Carry Output = 0
Thus, 4 bit AND Output = 1111

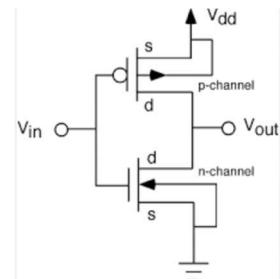
NGSPICE IMPLEMENTATION

Ngspice implementation was similar to Verilog Structural Code implementation, just with the implementation of Gates also being done using their transistor circuits. So here we get the freedom to choose the Width and Length parameters for the transistor to make gates.

```
.include TSMC_180nm.txt  
.param SUPPLY = 1.8  
.param  $\lambda$  = 0.18u  
  
.param wn1 = {10* $\lambda$ }  
.param wn2 = {10* $\lambda$ }  
.param ln1 = {2* $\lambda$ }  
.param ln2 = {2* $\lambda$ }  
  
.param wp1 = wn1  
.param wp2 = wn1  
.param lp1 = { $\lambda$ }  
.param lp2 = { $\lambda$ }  
  
.global GND
```

Initial parameters ,
length and width for PMOS
and NMOS

```
.subckt NOT IN OUT VDD GND  
  
Mn3 OUT IN GND GND CMOSN W = {wn1} L = {ln1}  
+ AS = {5*wn1* $\lambda$ } PS = {10* $\lambda$  + 2*wn1} AD = {5*wn1* $\lambda$ } PD = {10* $\lambda$  + 2*wn1}  
  
Mp3 OUT IN VDD VDD CMOSP W = {wp1} L = {lp1}  
+ AS = {5*wp1* $\lambda$ } PS = {10* $\lambda$  + 2*wp1} AD = {5*wp1* $\lambda$ } PD = {10* $\lambda$  + 2*wp1}  
  
.ends NOT
```



In the subcircuit, the similar CMOS inverter circuit is implemented. The starting letter as 'M' denotes transistor and next inputs are the different terminals for NMOS and PMOS.
Similar subcircuits made for NAND and NOR gate.

Also, 'X' is used as first letter if you call these subcircuits. Thus other gates are made by calling these subcircuits
(The file with subcircuits require .sub filetype and not .cir or .ckt)

```

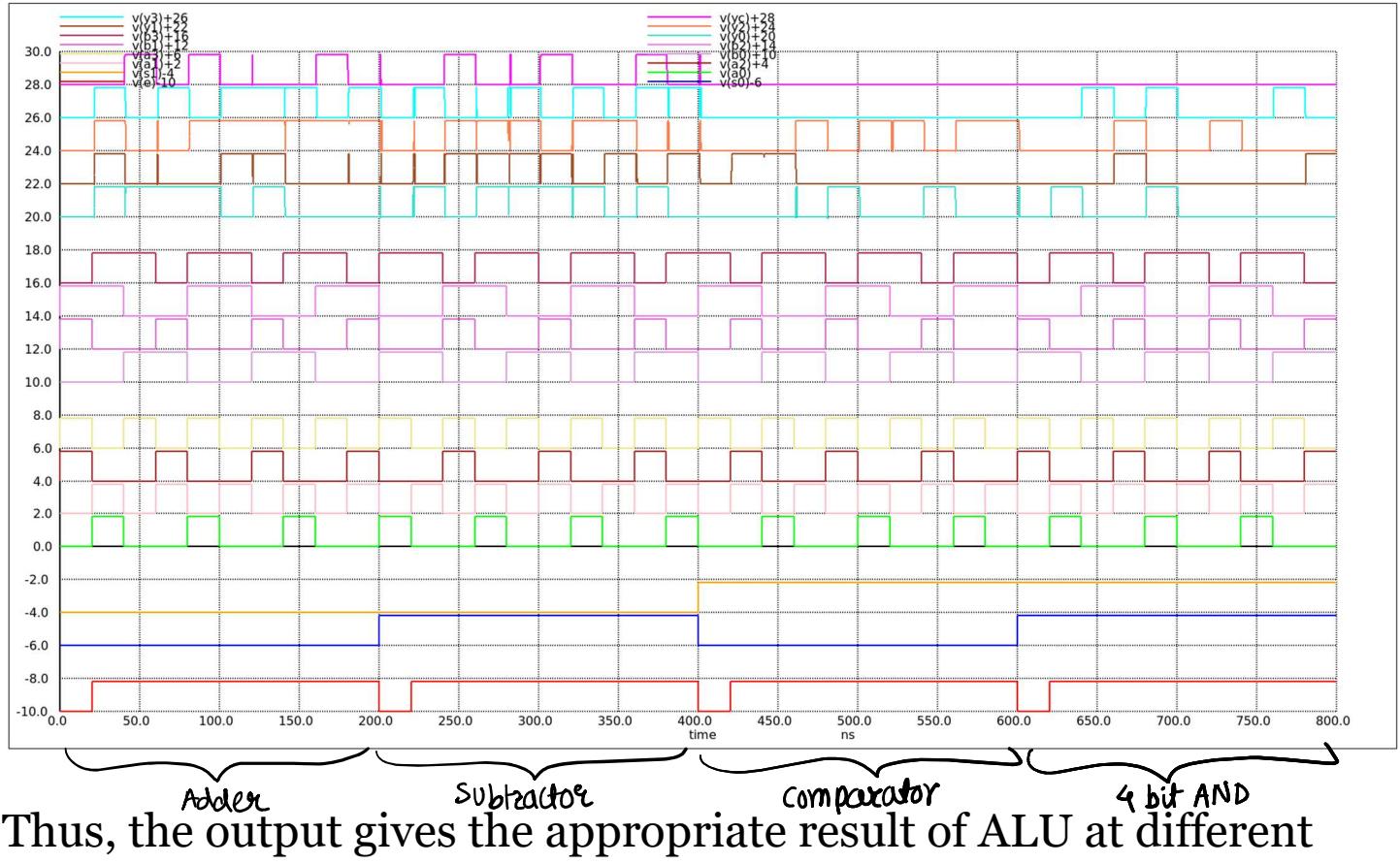
.subckt AND A B OUT VDD GND
    X1 A B t1 VDD GND NAND
    X2 t1 OUT VDD GND NOT
.ends AND

```

Similarly, all modules are made as subcircuits and then implemented into one ALU.cir file.
(.include required to use those subcircuits into the file)

So, in ALU.cir, a random input is given and output is plotted.
(Similar to testbench in Verilog)

NGSPICE OUTPUT :

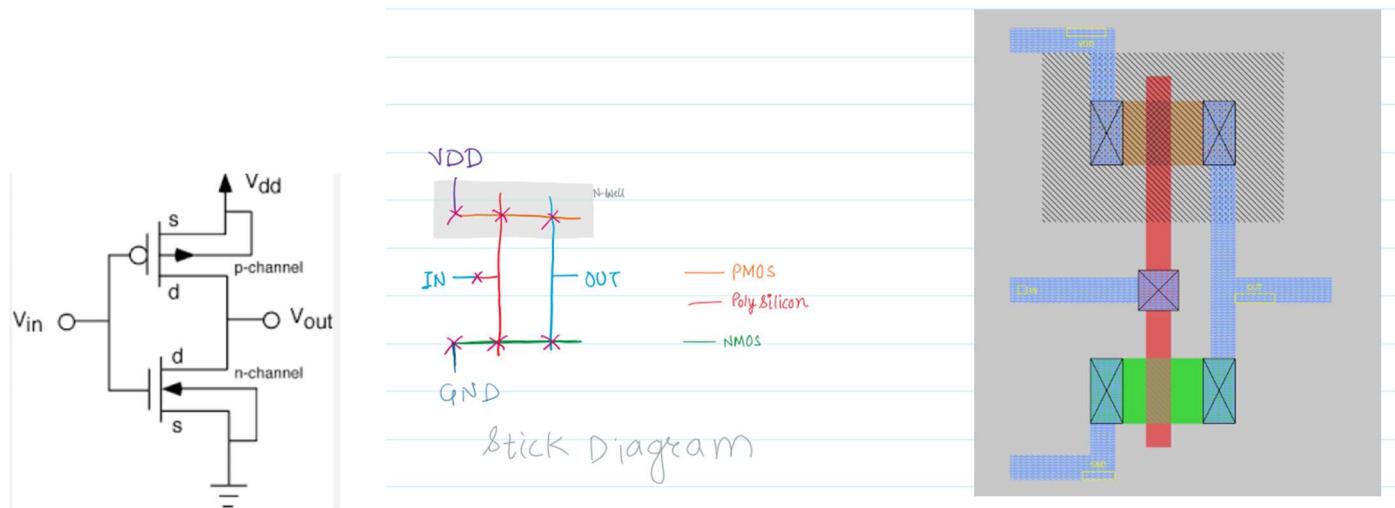


Thus, the output gives the appropriate result of ALU at different stages of Select lines inputs.

→ Issue : Ngspice was case-insensitive , so case change didn't make a new variable , but would denote the same variable

MAGIC LAYOUT IMPLEMENTATION

The Magic Layout implementation of the circuit was the most important part of the circuit. This part not only didn't give freedom to have length and width of transistor, but you actually design the transistor as on chip. The actual polysilicon and metals to be connected using different contacts.



“Magic allows designers to specify these diffusion regions and create connections between them using metal layers. The tool helps you visualize and create the layout of your integrated circuits based on the design rules and constraints of the chosen CMOS technology.”

The stick diagram gives us an idea of layout implementation of the gates. So here also, you initially make all the necessary gates, and then use them repeatedly to make the circuit.

We are using the 0.09 μ m technology for the MAGIC layout.

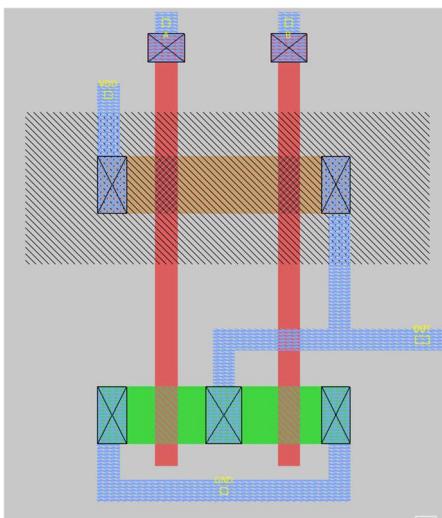
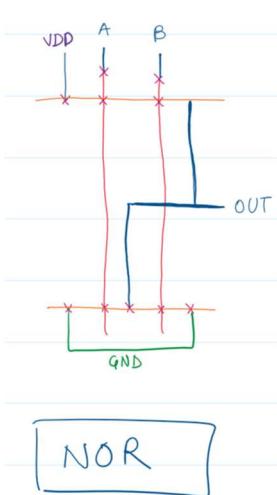
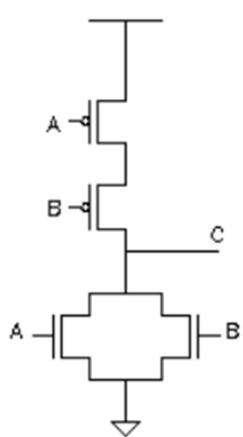
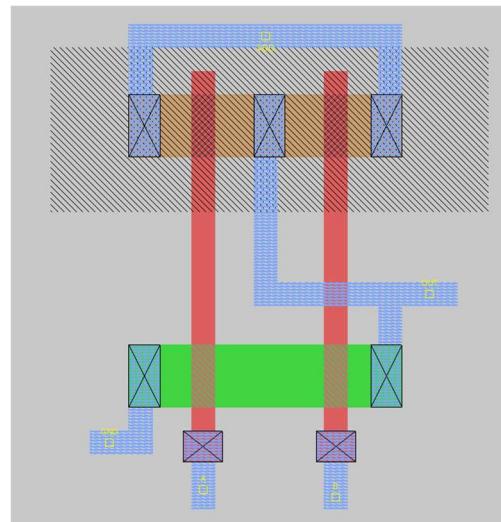
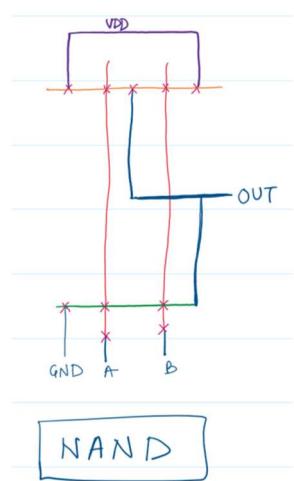
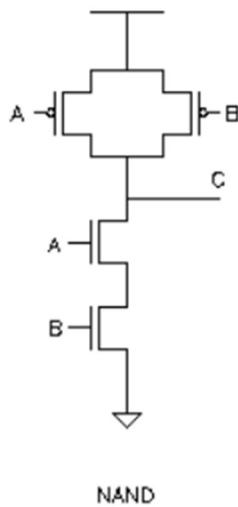
We consider the p substrate (pwell) by default in the layout, but then for PMOS, we need to make the nwell. The ndiffusion and pdiffusion are used for Source and Drain, and the polysilicon acts as the Gate of the transistor.

Challenges faced:

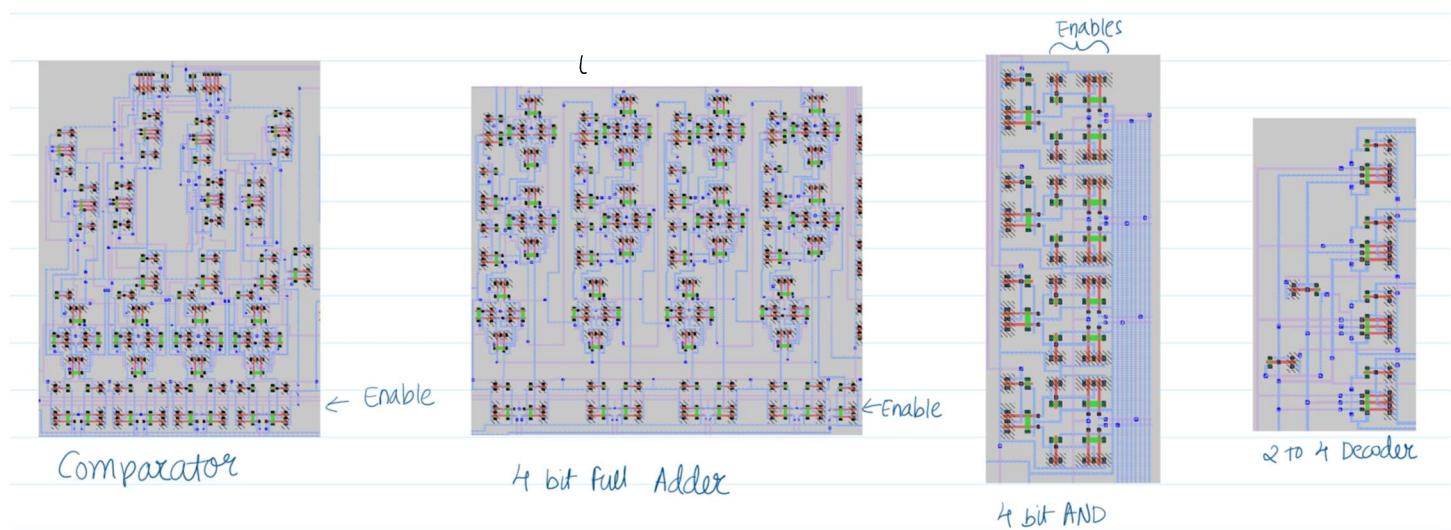
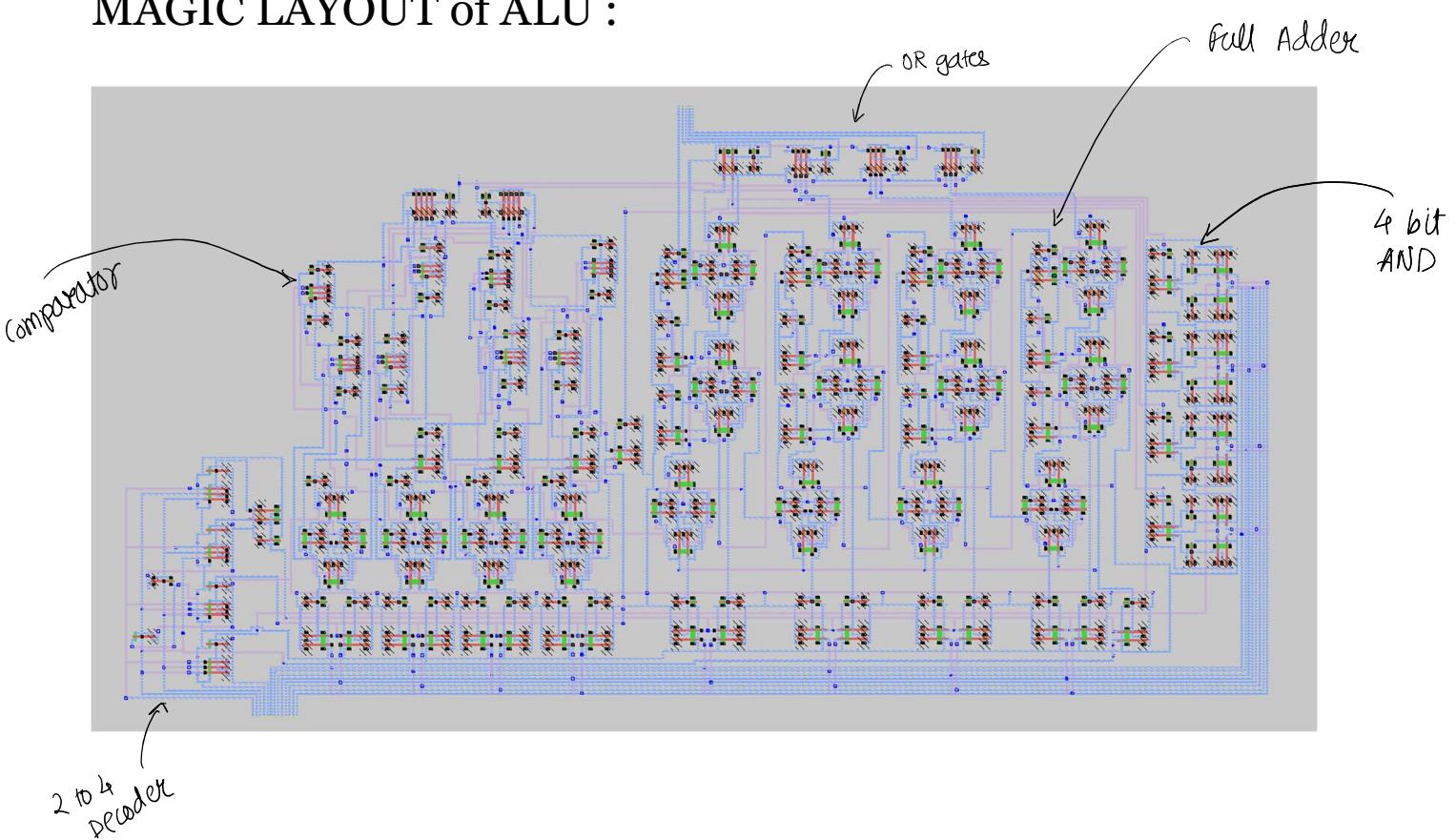
- The actual connections for all gates i.e. its VDD, GND, input and output are needed to be connected using metal. So actually you have VDD and GND interconnected to all gates. These things were challenging.

So the complex connections were solved using metal1, metal2 (different materials) which doesn't connect unless a contact is used.

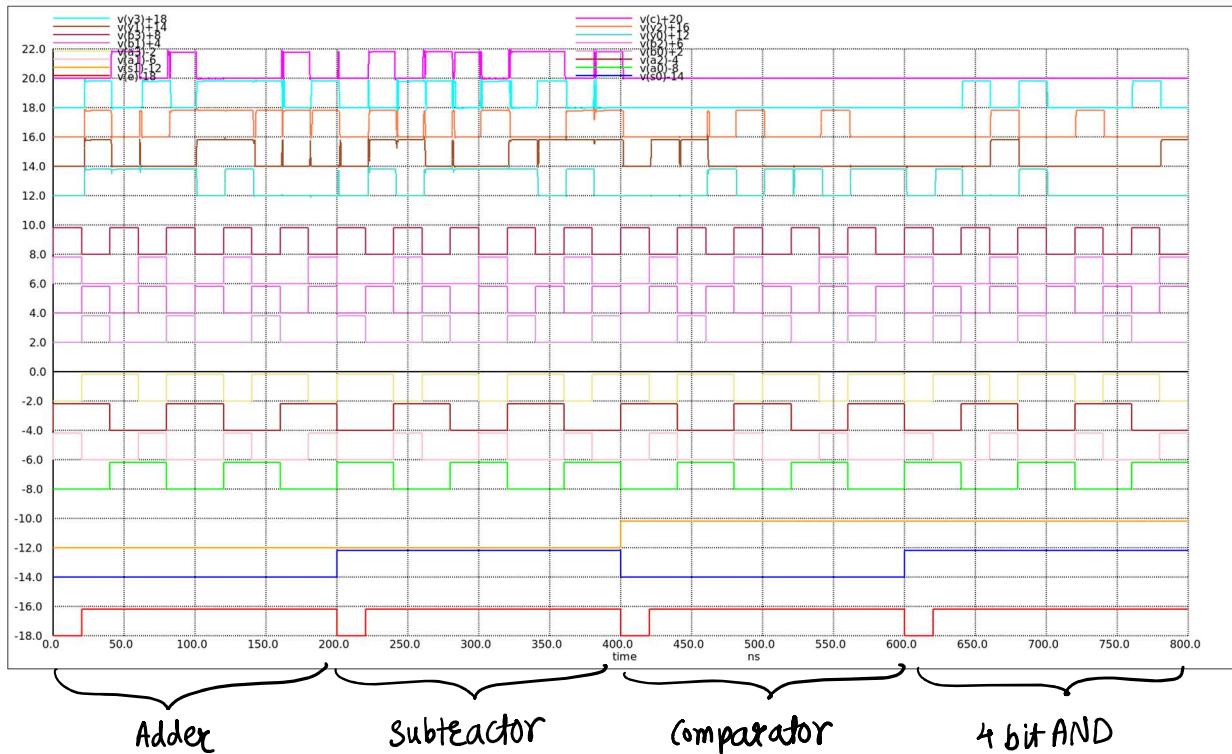
- Keeping the whole circuit connected using different contacts felt tough enough



MAGIC LAYOUT of ALU :



Then, we extract the magic file into .ext and then to .spice file. After required modifications, (nfet->CMOSN, pfet->CMOSP), adding input, plotting all the parameters, we get this output from NGSPICE for arbitrary pulse inputs :



∴ The desired output is obtained, and hence the Layout based Spice file works properly.

DELAY ANALYSIS :

After obtaining the spice file for the magic layout (converted to .cir), we perform delay analysis upon that file.

Thus, we give the desired input such that we get the change in every output with change in input bits. This is to be performed for all input output combinations.

Considering for each operation separately, we have delay analysis for all 4 combinations of select lines separately. Then we would check for maximum delay and get the critical path.

So, we need to figure out, the perfect input combinations to get the required delays

For Adder,
the preferred input is

$A = A_3 A_2 A_1 A_0 = \text{Pulse input}$, $B = B_3 B_2 B_1 B_0 = \text{DC value } 0$

$$\left\{ \begin{array}{l} 0000 + 0000 = 0000 \\ \downarrow \qquad \downarrow \\ 1111 + 0000 = 1111 \end{array} \right\}; \quad \begin{matrix} \text{Rise} \rightarrow \text{Rise} \\ A \rightarrow 4 \end{matrix}, \quad \begin{matrix} \text{Fall} \rightarrow \text{Fall} \\ A \rightarrow 4 \end{matrix}$$

similarly for calculating delay of inputs of B with outputs

$A = A_3 A_2 A_1 A_0 = \text{DC } 0$, $B = B_3 B_2 B_1 B_0 = \text{Pulse input}$

$$\left\{ \begin{array}{l} 0000 + 0000 = 0000 \\ \downarrow \qquad \downarrow \\ 0000 + 1111 = 1111 \end{array} \right\}; \quad \begin{matrix} \text{Rise} \rightarrow \text{Rise} \\ B \rightarrow 4 \end{matrix}, \quad \begin{matrix} \text{Fall} \rightarrow \text{Fall} \\ B \rightarrow 4 \end{matrix}$$

Similarly, for Subtractor,

$A = A_3A_2A_1A_0$ = Pulse input, $B = B_3B_2B_1B_0$ = DC value 0

$$\left\{ \begin{array}{l} 0000 - 0000 = 0000 \\ 1111 - 0000 = 1111 \end{array} \right\}; \begin{array}{l} \text{Rise} \rightarrow \text{Rise}, \text{fall} \rightarrow \text{fall} \\ A \rightarrow Y \qquad \qquad \qquad A \rightarrow Y \end{array}$$

$A = A_3A_2A_1A_0$ = DC 1, $B = B_3B_2B_1B_0$ = Pulse input

$$\left\{ \begin{array}{l} 1111 - 0000 = 1111 \\ 1111 - 1111 = 0000 \end{array} \right\}; \begin{array}{l} \text{Rise} \rightarrow \text{fall}, \text{fall} \rightarrow \text{rise} \\ B \rightarrow Y \qquad \qquad \qquad B \rightarrow Y \end{array}$$

Similarly, for 4 bit AND

$A = A_3A_2A_1A_0$ = Pulse input, $B = B_3B_2B_1B_0$ = DC value 1

$$\left\{ \begin{array}{l} 0000 \cdot 1111 = 0000 \\ 1111 \cdot 1111 = 1111 \end{array} \right\}; \begin{array}{l} \text{Rise} \rightarrow \text{Rise}, \text{fall} \rightarrow \text{fall} \\ A \rightarrow Y \qquad \qquad \qquad A \rightarrow Y \end{array}$$

$A = A_3A_2A_1A_0$ = DC 1, $B = B_3B_2B_1B_0$ = Pulse input

$$\left\{ \begin{array}{l} 1111 \cdot 0000 = 0000 \\ 1111 \cdot 1111 = 1111 \end{array} \right\}; \begin{array}{l} \text{Rise} \rightarrow \text{fall}, \text{fall} \rightarrow \text{rise} \\ B \rightarrow Y \qquad \qquad \qquad B \rightarrow Y \end{array}$$

Now, for Comparator, we have different cases for all 3 different cases of outputs.

$A > B$

$A = A_3 A_2 A_1 A_0 = \text{Pulse input}$, }
 $B = B_3 B_2 B_1 B_0 = \text{DC } 0$, }
Rise \rightarrow Rise, Fall \rightarrow Fall
 $A \rightarrow Y$ $A \rightarrow Y$

$A = A_3 A_2 A_1 A_0 = \text{DC } 1$, }
 $B = B_3 B_2 B_1 B_0 = \text{Pulse input}$, }
Fall \rightarrow Rise, Rise \rightarrow Fall
 $B \rightarrow Y$ $B \rightarrow Y$

$A = B$

$A = A_3 A_2 A_1 A_0 = \text{Pulse input}$, }
 $B = B_3 B_2 B_1 B_0 = \text{DC } 1$, }
Rise \rightarrow Rise, Fall \rightarrow Fall
 $A \rightarrow Y$ $A \rightarrow Y$

$A = A_3 A_2 A_1 A_0 = \text{DC } 1$, }
 $B = B_3 B_2 B_1 B_0 = \text{Pulse input}$, }
Rise \rightarrow Rise, Fall \rightarrow Fall
 $B \rightarrow Y$ $B \rightarrow Y$

$A < B$

$A = A_3 A_2 A_1 A_0 = \text{Pulse input}$, }
 $B = B_3 B_2 B_1 B_0 = \text{DC } 1$, }
Rise \rightarrow Fall, Fall \rightarrow Rise
 $A \rightarrow Y$ $A \rightarrow Y$

$A = A_3 A_2 A_1 A_0 = \text{DC } 0$, }
 $B = B_3 B_2 B_1 B_0 = \text{Pulse input}$, }
Rise \rightarrow Rise, Fall \rightarrow Fall
 $B \rightarrow Y$ $B \rightarrow Y$

DELAY ANALYSIS FOR IDEAL NGSPICE CIRCUIT (PRE-LAYOUT)

```
SPICE_DELAY > E ADDER_Delay_output.txt
1 Delay Analysis for ADDER :
2
3 tpd = 8.61042e-10 >> Input = A0 , Output = Y0
4 tpd = 8.61041e-10 >> Input = A0 , Output = Y1
5 tpd = 8.61041e-10 >> Input = A0 , Output = Y2
6 tpd = 8.47114e-10 >> Input = A0 , Output = Y3
7
8 tpd = 8.61042e-10 >> Input = A1 , Output = Y0
9 tpd = 8.61041e-10 >> Input = A1 , Output = Y1
10 tpd = 8.61041e-10 >> Input = A1 , Output = Y2
11 tpd = 8.47114e-10 >> Input = A1 , Output = Y3
12
13 tpd = 8.61042e-10 >> Input = A2 , Output = Y0
14 tpd = 8.61041e-10 >> Input = A2 , Output = Y1
15 tpd = 8.61041e-10 >> Input = A2 , Output = Y2
16 tpd = 8.47114e-10 >> Input = A2 , Output = Y3
17
18 tpd = 8.61042e-10 >> Input = A3 , Output = Y0
19 tpd = 8.61041e-10 >> Input = A3 , Output = Y1
20 tpd = 8.61041e-10 >> Input = A3 , Output = Y2
21 tpd = 8.47114e-10 >> Input = A3 , Output = Y3
22
23 tpd = 1.10301e-09 >> Input = B0 , Output = Y0
24 tpd = 1.10301e-09 >> Input = B0 , Output = Y1
25 tpd = 1.10301e-09 >> Input = B0 , Output = Y2
26 tpd = 1.09164e-09 >> Input = B0 , Output = Y3
27
28 tpd = 1.10301e-09 >> Input = B1 , Output = Y0
29 tpd = 1.10301e-09 >> Input = B1 , Output = Y1
30 tpd = 1.10301e-09 >> Input = B1 , Output = Y2
31 tpd = 1.09164e-09 >> Input = B1 , Output = Y3
32
33 tpd = 1.10301e-09 >> Input = B2 , Output = Y0
34 tpd = 1.10301e-09 >> Input = B2 , Output = Y1
35 tpd = 1.10301e-09 >> Input = B2 , Output = Y2
36 tpd = 1.09164e-09 >> Input = B2 , Output = Y3
37
38 tpd = 1.10301e-09 >> Input = B3 , Output = Y0
39 tpd = 1.10301e-09 >> Input = B3 , Output = Y1
40 tpd = 1.10301e-09 >> Input = B3 , Output = Y2
41 tpd = 1.09164e-09 >> Input = B3 , Output = Y3
42
```

Adder (So = o, S1 = o)

```
SPICE_DELAY > E SUBTRACTOR_Delay_output.txt
1 Delay Analysis for SUBTRACTOR :
2
3 tpd = 1.15503e-09 >> Input = A0 , Output = Y0
4 tpd = 1.36279e-09 >> Input = A0 , Output = Y1
5 tpd = 1.56579e-09 >> Input = A0 , Output = Y2
6 tpd = 1.76029e-09 >> Input = A0 , Output = Y3
7
8 tpd = 1.15503e-09 >> Input = A1 , Output = Y0
9 tpd = 1.36279e-09 >> Input = A1 , Output = Y1
10 tpd = 1.56579e-09 >> Input = A1 , Output = Y2
11 tpd = 1.76029e-09 >> Input = A1 , Output = Y3
12
13 tpd = 1.15503e-09 >> Input = A2 , Output = Y0
14 tpd = 1.36279e-09 >> Input = A2 , Output = Y1
15 tpd = 1.56579e-09 >> Input = A2 , Output = Y2
16 tpd = 1.76029e-09 >> Input = A2 , Output = Y3
17
18 tpd = 1.15503e-09 >> Input = A3 , Output = Y0
19 tpd = 1.36279e-09 >> Input = A3 , Output = Y1
20 tpd = 1.56579e-09 >> Input = A3 , Output = Y2
21 tpd = 1.76029e-09 >> Input = A3 , Output = Y3
22
23 tpd = 1.59956e-09 >> Input = B0 , Output = Y0
24 tpd = 1.80922e-09 >> Input = B0 , Output = Y1
25 tpd = 2.01105e-09 >> Input = B0 , Output = Y2
26 tpd = 2.19942e-09 >> Input = B0 , Output = Y3
27
28 tpd = 1.59956e-09 >> Input = B1 , Output = Y0
29 tpd = 1.80922e-09 >> Input = B1 , Output = Y1
30 tpd = 2.01105e-09 >> Input = B1 , Output = Y2
31 tpd = 2.19942e-09 >> Input = B1 , Output = Y3
32
33 tpd = 1.59956e-09 >> Input = B2 , Output = Y0
34 tpd = 1.80922e-09 >> Input = B2 , Output = Y1
35 tpd = 2.01105e-09 >> Input = B2 , Output = Y2
36 tpd = 2.19942e-09 >> Input = B2 , Output = Y3
37
38 tpd = 1.59956e-09 >> Input = B3 , Output = Y0
39 tpd = 1.80922e-09 >> Input = B3 , Output = Y1
40 tpd = 2.01105e-09 >> Input = B3 , Output = Y2
41 tpd = 2.19942e-09 >> Input = B3 , Output = Y3
42
```

Subtractor (So = 1, S1 = o)

```
SPICE_DELAY > E COMPARATOR_Delay_output.txt
1 Delay Analysis for COMPARATOR :
2
3 tpd = 1.09891e-09 >> Input = A0 , Output = Y0 (A>B)
4 tpd = 1.09891e-09 >> Input = A1 , Output = Y0 (A>B)
5 tpd = 1.09891e-09 >> Input = A2 , Output = Y0 (A>B)
6 tpd = 1.09891e-09 >> Input = A3 , Output = Y0 (A>B)
7
8 tpd = 1.18506e-09 >> Input = B0 , Output = Y0 (A>B)
9 tpd = 1.18506e-09 >> Input = B1 , Output = Y0 (A>B)
10 tpd = 1.18506e-09 >> Input = B2 , Output = Y0 (A>B)
11 tpd = 1.18506e-09 >> Input = B3 , Output = Y0 (A>B)
12
13 tpd = 1.58157e-09 >> Input = A0 , Output = Y1 (A=B)
14 tpd = 1.58157e-09 >> Input = A1 , Output = Y1 (A=B)
15 tpd = 1.58157e-09 >> Input = A2 , Output = Y1 (A=B)
16 tpd = 1.58157e-09 >> Input = A3 , Output = Y1 (A=B)
17
18 tpd = 1.63507e-09 >> Input = B0 , Output = Y1 (A=B)
19 tpd = 1.63507e-09 >> Input = B1 , Output = Y1 (A=B)
20 tpd = 1.63507e-09 >> Input = B2 , Output = Y1 (A=B)
21 tpd = 1.63507e-09 >> Input = B3 , Output = Y1 (A=B)
22
23 tpd = 1.15360e-09 >> Input = A0 , Output = Y2 (A<B)
24 tpd = 1.15360e-09 >> Input = A1 , Output = Y2 (A<B)
25 tpd = 1.15360e-09 >> Input = A2 , Output = Y2 (A<B)
26 tpd = 1.15360e-09 >> Input = A3 , Output = Y2 (A<B)
27
28 tpd = 1.12569e-09 >> Input = B0 , Output = Y2 (A<B)
29 tpd = 1.12569e-09 >> Input = B1 , Output = Y2 (A<B)
30 tpd = 1.12569e-09 >> Input = B2 , Output = Y2 (A<B)
31 tpd = 1.12569e-09 >> Input = B3 , Output = Y2 (A<B)
32
```

Comparator (So = 1, S1 = o)

```
SPICE_DELAY > E 4BIT_AND_Delay_output.txt
1 Delay Analysis for 4 BIT AND :
2
3 tpd = 6.61202e-10 >> Input = A0 , Output = Y0
4 tpd = 6.61202e-10 >> Input = A0 , Output = Y1
5 tpd = 6.61202e-10 >> Input = A0 , Output = Y2
6 tpd = 4.93219e-10 >> Input = A0 , Output = Y3
7
8 tpd = 6.61202e-10 >> Input = A1 , Output = Y0
9 tpd = 6.61202e-10 >> Input = A1 , Output = Y1
10 tpd = 6.61202e-10 >> Input = A1 , Output = Y2
11 tpd = 4.93219e-10 >> Input = A1 , Output = Y3
12
13 tpd = 6.61202e-10 >> Input = A2 , Output = Y0
14 tpd = 6.61202e-10 >> Input = A2 , Output = Y1
15 tpd = 6.61202e-10 >> Input = A2 , Output = Y2
16 tpd = 4.93219e-10 >> Input = A2 , Output = Y3
17
18 tpd = 6.61202e-10 >> Input = A3 , Output = Y0
19 tpd = 6.61202e-10 >> Input = A3 , Output = Y1
20 tpd = 6.61202e-10 >> Input = A3 , Output = Y2
21 tpd = 4.93219e-10 >> Input = A3 , Output = Y3
22
23 tpd = 6.84867e-10 >> Input = B0 , Output = Y0
24 tpd = 6.84867e-10 >> Input = B0 , Output = Y1
25 tpd = 6.84867e-10 >> Input = B0 , Output = Y2
26 tpd = 5.16688e-10 >> Input = B0 , Output = Y3
27
28 tpd = 6.84867e-10 >> Input = B1 , Output = Y0
29 tpd = 6.84867e-10 >> Input = B1 , Output = Y1
30 tpd = 6.84867e-10 >> Input = B1 , Output = Y2
31 tpd = 5.16688e-10 >> Input = B1 , Output = Y3
32
33 tpd = 6.84867e-10 >> Input = B2 , Output = Y0
34 tpd = 6.84867e-10 >> Input = B2 , Output = Y1
35 tpd = 6.84867e-10 >> Input = B2 , Output = Y2
36 tpd = 5.16688e-10 >> Input = B2 , Output = Y3
37
38 tpd = 6.84867e-10 >> Input = B3 , Output = Y0
39 tpd = 6.84867e-10 >> Input = B3 , Output = Y1
40 tpd = 6.84867e-10 >> Input = B3 , Output = Y2
41 tpd = 5.16688e-10 >> Input = B3 , Output = Y3
42
```

4 bit AND (So = 1, S1 = 1)

DELAY ANALYSIS FOR NGSPICE CIRCUIT EXTRACTED FROM MAGIC (POST-LAYOUT)

DELAY ANALYSIS > E ADDER_Delay_output.txt

```

1 Delay Analysis for ADDER :
2
3 tpd      = 1.06154e-09 >> Input = A0 , Output = Y0
4 tpd      = 1.07017e-09 >> Input = A0 , Output = Y1
5 tpd      = 1.06863e-09 >> Input = A0 , Output = Y2
6 tpd      = 1.04100e-09 >> Input = A0 , Output = Y3
7
8 tpd      = 1.06154e-09 >> Input = A1 , Output = Y0
9 tpd      = 1.07017e-09 >> Input = A1 , Output = Y1
10 tpd     = 1.06863e-09 >> Input = A1 , Output = Y2
11 tpd     = 1.04100e-09 >> Input = A1 , Output = Y3
12
13 tpd     = 1.06154e-09 >> Input = A2 , Output = Y0
14 tpd     = 1.07017e-09 >> Input = A2 , Output = Y1
15 tpd     = 1.06863e-09 >> Input = A2 , Output = Y2
16 tpd     = 1.04100e-09 >> Input = A2 , Output = Y3
17
18 tpd     = 1.06154e-09 >> Input = A3 , Output = Y0
19 tpd     = 1.07017e-09 >> Input = A3 , Output = Y1
20 tpd     = 1.06863e-09 >> Input = A3 , Output = Y2
21 tpd     = 1.04100e-09 >> Input = A3 , Output = Y3
22
23 tpd     = 1.02531e-09 >> Input = B0 , Output = Y0
24 tpd     = 1.03106e-09 >> Input = B0 , Output = Y1
25 tpd     = 1.02193e-09 >> Input = B0 , Output = Y2
26 tpd     = 9.90812e-10 >> Input = B0 , Output = Y3
27
28 tpd     = 1.02531e-09 >> Input = B1 , Output = Y0
29 tpd     = 1.03106e-09 >> Input = B1 , Output = Y1
30 tpd     = 1.02193e-09 >> Input = B1 , Output = Y2
31 tpd     = 9.90812e-10 >> Input = B1 , Output = Y3
32
33 tpd     = 1.02531e-09 >> Input = B2 , Output = Y0
34 tpd     = 1.03106e-09 >> Input = B2 , Output = Y1
35 tpd     = 1.02193e-09 >> Input = B2 , Output = Y2
36 tpd     = 9.90812e-10 >> Input = B2 , Output = Y3
37
38 tpd     = 1.02531e-09 >> Input = B3 , Output = Y0
39 tpd     = 1.03106e-09 >> Input = B3 , Output = Y1
40 tpd     = 1.02193e-09 >> Input = B3 , Output = Y2
41 tpd     = 9.90812e-10 >> Input = B3 , Output = Y3
42

```

Adder (So = o, S1 = o)

DELAY ANALYSIS > E SUBTRACTOR_Delay_output.txt

```

1 Delay Analysis for SUBTRACTOR :
2
3 tpd      = 1.47361e-09 >> Input = A0 , Output = Y0
4 tpd      = 1.89895e-09 >> Input = A0 , Output = Y1
5 tpd      = 2.29117e-09 >> Input = A0 , Output = Y2
6 tpd      = 1.39813e-09 >> Input = A0 , Output = Y3
7
8 tpd      = 1.47361e-09 >> Input = A1 , Output = Y0
9 tpd      = 1.89895e-09 >> Input = A1 , Output = Y1
10 tpd     = 2.29117e-09 >> Input = A1 , Output = Y2
11 tpd     = 1.39813e-09 >> Input = A1 , Output = Y3
12
13 tpd     = 1.47361e-09 >> Input = A2 , Output = Y0
14 tpd     = 1.89895e-09 >> Input = A2 , Output = Y1
15 tpd     = 2.29117e-09 >> Input = A2 , Output = Y2
16 tpd     = 1.39813e-09 >> Input = A2 , Output = Y3
17
18 tpd     = 1.47361e-09 >> Input = A3 , Output = Y0
19 tpd     = 1.89895e-09 >> Input = A3 , Output = Y1
20 tpd     = 2.29117e-09 >> Input = A3 , Output = Y2
21 tpd     = 1.39813e-09 >> Input = A3 , Output = Y3
22
23 tpd     = 1.23874e-09 >> Input = B0 , Output = Y0
24 tpd     = 1.22659e-09 >> Input = B0 , Output = Y1
25 tpd     = 1.21877e-09 >> Input = B0 , Output = Y2
26 tpd     = 1.18331e-09 >> Input = B0 , Output = Y3
27
28 tpd     = 1.23874e-09 >> Input = B1 , Output = Y0
29 tpd     = 1.22659e-09 >> Input = B1 , Output = Y1
30 tpd     = 1.21877e-09 >> Input = B1 , Output = Y2
31 tpd     = 1.18331e-09 >> Input = B1 , Output = Y3
32
33 tpd     = 1.23874e-09 >> Input = B2 , Output = Y0
34 tpd     = 1.22659e-09 >> Input = B2 , Output = Y1
35 tpd     = 1.21877e-09 >> Input = B2 , Output = Y2
36 tpd     = 1.18331e-09 >> Input = B2 , Output = Y3
37
38 tpd     = 1.23874e-09 >> Input = B3 , Output = Y0
39 tpd     = 1.22659e-09 >> Input = B3 , Output = Y1
40 tpd     = 1.21877e-09 >> Input = B3 , Output = Y2
41 tpd     = 1.18331e-09 >> Input = B3 , Output = Y3
42

```

Subtractor (So = 1, S1 = o)

DELAY ANALYSIS > E 4BIT_AND_Delay_output.txt

```

1 Delay Analysis for 4 BIT AND :
2
3 tpd      = 6.24154e-10 >> Input = A0 , Output = Y0
4 tpd      = 6.35254e-10 >> Input = A0 , Output = Y1
5 tpd      = 6.49606e-10 >> Input = A0 , Output = Y2
6 tpd      = 6.40244e-10 >> Input = A0 , Output = Y3
7
8 tpd      = 6.24154e-10 >> Input = A1 , Output = Y0
9 tpd      = 6.35254e-10 >> Input = A1 , Output = Y1
10 tpd     = 6.49606e-10 >> Input = A1 , Output = Y2
11 tpd     = 6.40244e-10 >> Input = A1 , Output = Y3
12
13 tpd     = 6.24154e-10 >> Input = A2 , Output = Y0
14 tpd     = 6.35254e-10 >> Input = A2 , Output = Y1
15 tpd     = 6.49606e-10 >> Input = A2 , Output = Y2
16 tpd     = 6.40244e-10 >> Input = A2 , Output = Y3
17
18 tpd     = 6.24154e-10 >> Input = A3 , Output = Y0
19 tpd     = 6.35254e-10 >> Input = A3 , Output = Y1
20 tpd     = 6.49606e-10 >> Input = A3 , Output = Y2
21 tpd     = 6.40244e-10 >> Input = A3 , Output = Y3
22
23 tpd     = 6.43858e-10 >> Input = B0 , Output = Y0
24 tpd     = 6.55750e-10 >> Input = B0 , Output = Y1
25 tpd     = 6.68333e-10 >> Input = B0 , Output = Y2
26 tpd     = 6.58743e-10 >> Input = B0 , Output = Y3
27
28 tpd     = 6.43858e-10 >> Input = B1 , Output = Y0
29 tpd     = 6.55750e-10 >> Input = B1 , Output = Y1
30 tpd     = 6.68333e-10 >> Input = B1 , Output = Y2
31 tpd     = 6.58743e-10 >> Input = B1 , Output = Y3
32
33 tpd     = 6.43858e-10 >> Input = B2 , Output = Y0
34 tpd     = 6.55750e-10 >> Input = B2 , Output = Y1
35 tpd     = 6.68333e-10 >> Input = B2 , Output = Y2
36 tpd     = 6.58743e-10 >> Input = B2 , Output = Y3
37
38 tpd     = 6.43858e-10 >> Input = B3 , Output = Y0
39 tpd     = 6.55750e-10 >> Input = B3 , Output = Y1
40 tpd     = 6.68333e-10 >> Input = B3 , Output = Y2
41 tpd     = 6.58743e-10 >> Input = B3 , Output = Y3
42

```

DELAY ANALYSIS > E COMPARATOR_Delay_output.txt

```

1 Delay Analysis for COMPARATOR :
2
3 tpd      = 1.33683e-09 >> Input = A0 , Output = Y0 (A>B)
4 tpd      = 1.33683e-09 >> Input = A1 , Output = Y0 (A>B)
5 tpd      = 1.33683e-09 >> Input = A2 , Output = Y0 (A>B)
6 tpd      = 1.33683e-09 >> Input = A3 , Output = Y0 (A>B)
7
8 tpd      = 1.44843e-09 >> Input = B0 , Output = Y0 (A>B)
9 tpd      = 1.44843e-09 >> Input = B1 , Output = Y0 (A>B)
10 tpd     = 1.44843e-09 >> Input = B2 , Output = Y0 (A>B)
11 tpd     = 1.44843e-09 >> Input = B3 , Output = Y0 (A>B)
12
13 tpd     = 1.97206e-09 >> Input = A0 , Output = Y1 (A=B)
14 tpd     = 1.97206e-09 >> Input = A1 , Output = Y1 (A=B)
15 tpd     = 1.97206e-09 >> Input = A2 , Output = Y1 (A=B)
16 tpd     = 1.97206e-09 >> Input = A3 , Output = Y1 (A=B)
17
18 tpd     = 1.95243e-09 >> Input = B0 , Output = Y1 (A=B)
19 tpd     = 1.95243e-09 >> Input = B1 , Output = Y1 (A=B)
20 tpd     = 1.95243e-09 >> Input = B2 , Output = Y1 (A=B)
21 tpd     = 1.95243e-09 >> Input = B3 , Output = Y1 (A=B)
22
23 tpd     = 1.38295e-09 >> Input = A0 , Output = Y2 (A<B)
24 tpd     = 1.38295e-09 >> Input = A1 , Output = Y2 (A<B)
25 tpd     = 1.38295e-09 >> Input = A2 , Output = Y2 (A<B)
26 tpd     = 1.38295e-09 >> Input = A3 , Output = Y2 (A<B)
27
28 tpd     = 1.22784e-09 >> Input = B0 , Output = Y2 (A<B)
29 tpd     = 1.22784e-09 >> Input = B1 , Output = Y2 (A<B)
30 tpd     = 1.22784e-09 >> Input = B2 , Output = Y2 (A<B)
31 tpd     = 1.22784e-09 >> Input = B3 , Output = Y2 (A<B)
32

```

Comparator (So = 1, S1 = o)

→ The max. delay for Pre-Layout case was obtained from 'Subtractor'

∴ Critical path is obtained from 'Subtractor'

While overall delays are maximum in comparator part.

→ Similar observations could be obtained from the Post-Layout delay analysis ...